Analysis of nonlinearities in RF front-end architectures using a modified Volterra series approach

Michaël Goffioul, Piet Wambacq, Gerd Vandersteen, Stéphane Donnay
IMEC vzw, DESICS-MIRA
Kapeldreef 75, B-3001 Leuven, Belgium

Abstract

RF front-end architectures of today's wireless applications need to meet tough requirements on nonlinear distortion to minimize unwanted effects such as crosstalk. An analysis of the nonlinear behavior of analog communication circuits or architectures is not straightforward. This paper presents a modified Volterra series approach to the simulation of nonlinear systems described at the architectural level. The total computed response is decomposed in its nonlinear contributions and the main nonlinearities can be identified. This yields a better insight into the system's nonlinear behavior and allows simplifications. The simplified system can then be simulated more efficiently. The implementation is only based on vector calculation to minimize the computation time, and has been applied to a complete 5 GHz WLAN receiver front-end.

1. Introduction

The complexity of wireless communication systems is constantly increasing, resulting in more and more blocks that usually show nonlinear behavior. However, the use of high-order constellations (e.g. QAM64) requires the link to be as linear as possible, with a limited amount of interference. It is therefore important to understand correctly the nonlinear behavior of communication architectures, and to be able to identify the predominant nonlinearities in the system. Indeed actual systems contain a lot of such nonlinearities, but usually only a small part is important in the output response. The understanding of the overall nonlinear behavior can lead to better specifications at the circuit level.

Although there exist efficient methods for system simulation [1], the main goal is usually the efficiency, and they rarely provide a good insight into the nonlinear behavior of the system, especially the influence of individual nonlinearities on the system's overall performances like IP_3. On the other hand, such a tool exists for the circuit level [2]. This paper presents a tool to analyze the nonlinear behavior of RF front-end architectures containing standard blocks like amplifiers, mixers and filters. On one hand the formulation of the system into an equivalent circuit allows to take into account nonlinear behaviors that are difficult to introduce in other methods, like nonlinear loading effect or nonlinear feedback loops. On the other hand, the Volterra series analysis method based on single argument kernels, in combination with vector calculations, reduces significantly the memory consumption as well as the CPU time, and also allows to take into account frequency dependencies in the system's nonlinear blocks. As the method is based on Volterra series, all blocks must be represented by a power series, a Volterra series, a linear transfer function, or a combination of these. However because of the reduced CPU time, this method is able to compute very high-order responses – up to eleven, as in the example presented below – the only limitation being the CPU time. This allows in many cases a very good approximation of strongly nonlinear behaviors, such as the nonlinear behavior of a Gilbert cell mixer with respect to the LO signal. The use of other nonlinear circuit simulation methods like small-signal-large-signal analysis [3, 4] can also be used to decompose a strongly nonlinear block into a set of blocks that respect the limitations mentioned above.

The analysis method proposed in this paper decomposes the output response into a sum of contributions corresponding to the nonlinearity coefficients of the system's blocks. This decomposition allows to identify the most important coefficients, yielding a better understanding into the system's nonlinear behavior and possible simplifications. As the main target is not to build an efficient system simulator but more a tool to analyze the nonlinear behavior of communication systems, we can limit ourselves to simple input signals consisting of a few sinusoids. The simplified model resulting from our analysis tool usually contains only a few nonlinearities. This simplified model can then afterwards be excited by a realistic digital telecom signal. The CPU time of such simulation is much smaller because of the reduced complexity of the simplified system.

The second section details the analysis method, while the third section presents the application of this method to a complete 5 GHz WLAN receiver front-end architecture. The system is simulated up to order eleven and the simplification process results in only three main nonlinearities in baseband.

2. Description of the analysis method

The analysis method consists of a two-steps process. First the system is converted into an equivalent circuit which is then analyzed using a Volterra series based recursive algorithm, resulting in the decomposition of the output response into a sum of nonlinear contributions. Finally, these contributions are sorted and only the dominant ones are kept, leading to a simplified architecture with only a few nonlinearities.

2.1. Setup of equations

In a first step, the architecture under consideration is splitted into its linear and nonlinear part, so that the resulting system only consists of linear blocks and nonlinear sources and can be described by a matrix relation. For practical cases, the system's matrix is widely sparse. As an example, a system containing a 3rd order nonlinear amplifier followed by an ideal multiplier is represented by

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -k_1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} = \begin{bmatrix} V_{RF} \\ V_{LO} \\ i_1 \\ i_2 \end{bmatrix} \quad (1)$$

where i_1 and i_2 contain all nonlinear coefficients. The use of sparse matrix manipulation techniques greatly reduces the computation time as well as the storage size.

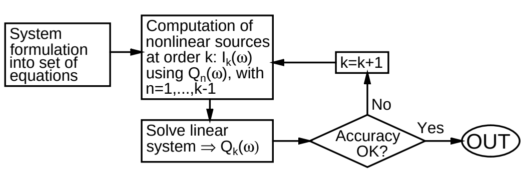

Figure 1. Simulation algorithm

2.2. Modified Volterra series approach

The simulation method is an adaptation at the system level of the modified Volterra series approach described in [5, 6]. The major drawback of most Volterra series based algorithms such as the one from [7] is the exponential growth with the order of the storage size of the multi-dimensional Fourier transforms of the Volterra kernels. Although this storage size can be reduced by the use of sparse matrices, this task becomes cumbersome when dealing with high-order kernels. On the other hand, the system simplification that is performed in a next step (see below) requires

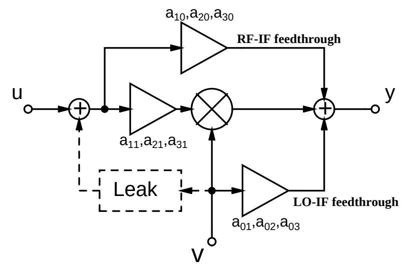

Figure 2. Mixer model

to compare real spectra, hence one needs to integrate those multi-dimensional kernels to reduce them to a single frequency argument [7, 8].

Both issues can be solved simultaneously by using the approach introduced in [5], where all kernels are integrated with respect to the frequency arguments. This integration is possible *a priori* for an input signal that consists of multiple sinusoidal signals (*multisine* signal), and simply reduces to a sum because of the properties of the Dirac impulse function. The algorithm remains the same as described in [7], except that it acts on single-argument functions, as shown on figure 1. To implement this algorithm, only the expressions of nonlinear current sources for an amplifier and an ideal multiplier are needed. Indeed, a real mixer described by

$$\begin{aligned} y = a_{10}u + a_{20}u^2 + a_{30}u^3 + a_{01}v + a_{02}v^2 + \\ a_{03}v^3 + a_{11}uv + a_{21}u^2v + a_{31}u^3v \end{aligned} \quad (2)$$

can be easily decomposed in a set of amplifiers, filters and multipliers as shown on figure 2. In this case, it consists of a nonlinear feedthrough path for the RF and the LO signal, and a central part including an amplifier and a multiplier, which corresponds to the core part of the mixer and takes into account its nonlinear behavior with respect to the RF signal. It is also possible to introduce LO-RF leakage as suggested on figure 2.

In the next paragraphs, we discuss how the response of an amplifier and a multiplier – modeled at high-level – is computed. We limit the discussion to static nonlinear behavior, although the method is applicable as well to dynamic nonlinear behavior.

2.2.1. Amplifier

For a static nonlinear amplifier with an input/output relation

$$w(t) = \sum_{n=1}^{\infty} k_n s(t)^n \quad (3)$$

the nonlinear current source of order n is given by the relation (4)(see next page)[5] with

$$S_{q_j}(\omega) = \sum_{r_j=-N_{(j)}}^{N_{(j)}} S_{q_j,r_j} \delta(\omega - \omega_{r_j}^{(j)}) \quad (6)$$

$$W_n(\omega) = \sum_{r=2}^{n} \Big\{ \sum_{q_1=1}^{a_1} \cdots \sum_{q_{r-1}=q_{r-2}}^{a_{r-1}} M k_r \Big[\sum_{r_1=-N_{(1)}}^{N_{(1)}} \cdots \sum_{r_r=-N_{(r)}}^{N_{(r)}} S_{q_1,r_1} \dots S_{q_r,r_r} \delta(\omega - \omega_{r_1}^{(1)} - \cdots - \omega_{r_r}^{(r)}) \Big] \Big\} \qquad (4)$$

$$W_n(\omega) = \sum_{k=1}^{n-1} \Big\{ \sum_{i=-N_{U_k}}^{N_{U_k}} \sum_{j=-N_{V_{n-k}}}^{N_{V_{n-k}}} K_m U_{k,i} V_{n-k,j} \delta(\omega - \omega_i^{U_k} - \omega_j^{V_{n-k}}) \Big\} \qquad (5)$$

$$a_i = \mathrm{Int}\left[\frac{n - \sum_{j=1}^{i-1} q_j}{r - i + 1}\right] \quad i = 1, 2, \dots, r \qquad (7)$$

$$q_r = a_r \qquad (8)$$

$$M = \frac{r!}{n_1! n_2! \dots n_{n-r+1}!} \qquad (9)$$

where $S_{q_j}(\omega)$ is the Fourier transform of $s(t)$ at order q_j, Int is the truncation function and n_i is the number of times one of the q_j with $j = 1, \dots, r$ is equal to i. For example, the expression of the nonlinear current source of order two and for two input sines at ω_1 and ω_2 with amplitude A_1 and A_2 respectively, is given by

$$W_2(\omega) = k_2 \frac{A_1 A_2}{4} \sum \delta(\omega \pm \omega_1 \pm \omega_2) \qquad (10)$$

where δ is the impulse function. The terminal impedances of the amplifier can be introduced in the form of additional equations in the matrix formulation of the system like in equation (1). Furthermore, an amplifier with dynamic nonlinearities can be decomposed into a combination of static nonlinearities and linear transfer functions.

2.2.2. Ideal multiplier

For an ideal multiplier with an input/output relation

$$w(t) = K_m u(t) v(t) \qquad (11)$$

the nonlinear current source expression can be obtained by using the extension of the Volterra series theory to multiple input, multiple output (MIMO) systems [9] in conjunction with an approach similar to the one used for amplifiers. This results in the expression (5), where $\omega_l^{X_p}$ is the l^{th} nonzero frequency component of x at order p, with x being either u or v. Note that a multiplier is essentially a nonlinear element and it increases the order of the wanted signal by one. The ideal multiplier, possibly in combination with terminal impedances, is the basic building block for a high-level model of a mixer.

2.3. Model simplification

Equation (4) shows that the nonlinear current sources consist of a sum of contributions, each of these contributions corresponding to a nonlinearity coefficient of the system. By computing these contributions individually, it is thus possible to decompose the output response. In a post-processing step, all contributions for each order are then sorted and compared, and only the most significant ones – up to a specified accuracy – are kept [2]. This comparison is made on the magnitude of the spectral components in a frequency band of interest. This simplification process reduces the system complexity since most often very few nonlinearities play a role. The decomposition of the output response also yields insight into the system's nonlinear behavior. Furthermore, the simplified architecture, containing only the most important nonlinearities, can now be simulated more efficiently.

2.4. Implementation

The simulation method described above has been implemented in MATLAB. In a straightforward, recursive implementation of the above expressions, the computation time is mainly governed by two tasks: the recursive sum over the nonzero spectral components in relations (4) and (5), and the solution of the linear circuit at each frequency.

The first task can be tackled efficiently by trying to use only vectors computations. For example the internal part of (4) can be calculated at once by using the Kronecker product and sum [10]. If F_j is the vector of nonzero frequency components of S_{q_j}, and I_j is the vector of corresponding indices, the resulting frequency components and indices are given by

$$F = F_1 \otimes F_2 \otimes \cdots \otimes F_r \qquad (12)$$

$$I = I_1 \oplus I_2 \oplus \cdots \oplus I_r \qquad (13)$$

The use of this technique reduces the computation time by a factor 10 compared to a recursive implementation. The presence of feedback loops in the system doesn't affect this vector computation based method as we are dealing with individual nonlinear current sources, hence the computation time is not increased.

The second task can be handled efficiently by taking advantage of the fact that the circuit is solved for each nonlinear contribution separately. This means that for each contribution, there is only one nonzero element in the right part of (1). Then it can be shown that for $n > 1$ the matrix $\mathbf{Q_n}(\omega)$ of the q_j for all frequencies is given by

$$\mathbf{Q_n}(\omega) = [\mathbf{c_k}(\omega_1) i_{k_n}(\omega_1) \dots \mathbf{c_k}(\omega_N) i_{k_n}(\omega_N)] \qquad (14)$$

where $\mathbf{c_k}$ is the k$^{\text{th}}$ column of the inverse of the admittance matrix describing the equivalent circuit associated with the system, and i_{k_n} is the nonlinear current source of order n at position k. The relation (14) can also be performed in a single matrix operation within MATLAB.

Finally, it is interesting to note that the expressions of the nonlinear current sources are implemented in a generic way, and not hard-coded, as it is usually the case in Volterra series based approaches [2]. In this case, it is possible to go up to an arbitrarily high order, the main limitation of the method being the CPU time.

3. Example: WLAN receiver front-end

The approach described above is illustrated with an analysis of a 5 GHz WLAN receiver front-end (figure 3). The system has been simulated up to order eleven using the technique described in the previous section. The excitation signal consists of five sinusoids with random phase, which is sufficient to highlight the system's nonlinear behavior. The result obtained for the total output response is shown in figure 4. In that figure, the response computed with the approach described is compared to the result obtained by classical straightforward computation where the linear blocks are handled in the frequency domain, and the nonlinear blocks are handled in the time domain. The agreement is nearly perfect showing a relative error in magnitude lower than 1%, thereby validating the method used.

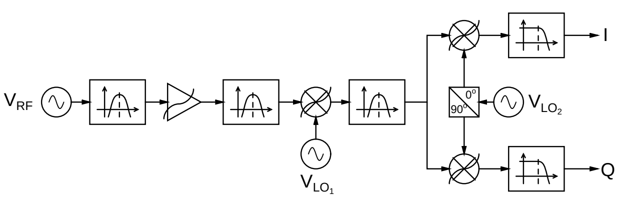

Figure 3. 5 GHz WLAN receiver front-end with a nonlinear LNA and nonlinear mixers

Figure 5 shows the decomposition up to order seven of the output signal in baseband. From that figure, the main nonlinear contributions are found to be the 3$^{\text{rd}}$ order coefficient k_3 of the LNA, as well as the 3$^{\text{rd}}$ order coefficient a_{31} of the RF and IF mixers. If the designer is only interested in baseband output signal for efficient simulations – with a realistic communication input signal – the system can be simplified to only contain those three nonlinearities. The term *linear* in figure 5 corresponds to the wanted output signal, that is linearly proportional to the input amplitude, although the intermediate mixers in the front-end increase its order. A comparison of the computation time between the complete and the simplified system is presented in figure 6. The system simplification decreases the CPU time by a factor of about five for the same accuracy. For longer examples with more nonlinearities, but still with just a few

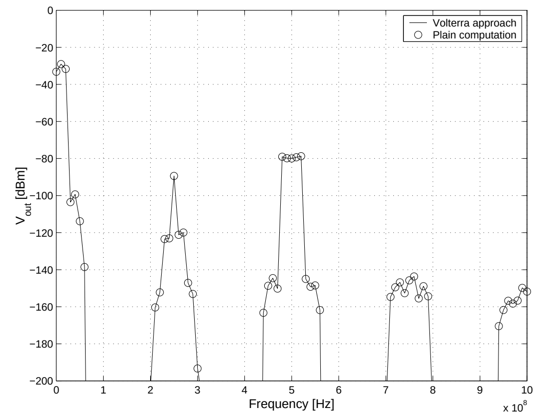

Figure 4. Total output response of the WLAN front-end for a multisine (5 sines) input signal up to order eleven.

dominant ones, the speedup is usually higher.

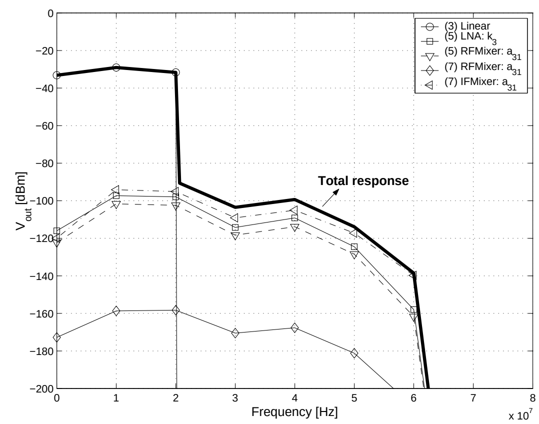

Figure 5. Nonlinear contributions in the baseband for all computed orders. The thick line corresponds to the total response. (n) means order n.

From the equations (4) and (5), it it easy to see that the number of operations, hence the computation time, grows exponentially with the considered order. Indeed at each step, the number of nonzero spectral components increases. The total computation time can still be maintained at a reasonable value by using vector calculation within MATLAB for the computation of the right-hand side of these two equations. However, it is possible to decrease further the CPU time by inserting a threshold value as follows. With increasing order, the magnitude of the newly created components decreases to soon become negligibly small. Therefore, spectral components with a magnitude below the specified threshold can be set to zero without significant loss of accuracy. Figure 7 shows the computation time versus order for two threshold values. It can be seen that a higher thresh-

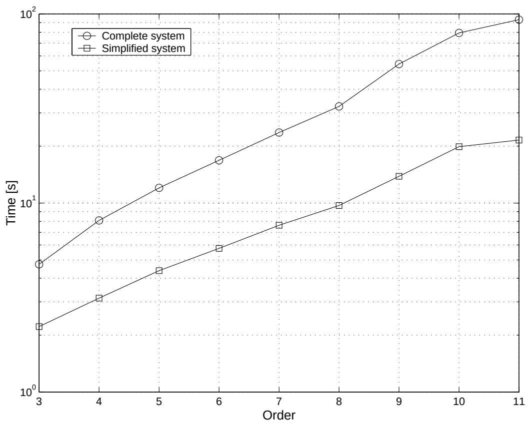

Figure 6. Comparison of computation time for the complete (◯) and the simplified (□) system

old leads to a significant decrease of computation time.

Figure 7 also shows the comparison of the CPU time for two operations in the total process. The nonlinear current sources computation time increases faster, and becomes predominant. This is due to the increase of the nonzero frequency components as shown by (4). The *data arrangement* is related to equation (13). The vector I contains indices larger than the number of samples N_s, hence the need to postprocess F and I to have only spectral components in the range $[0, f_{sampling}]$. The CPU time associated with this operation depends on the number of spectral components and thus increases in the same way as nonlinear current sources computation.

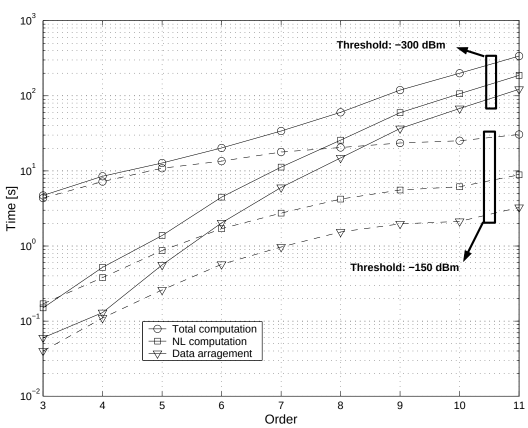

Figure 7. Analysis of computation time for the WLAN front-end. (◯) Total, (□) nonlinear current sources, (▽) data arrangement

4. Conclusions

To analyze the nonlinear behavior of architectures for digital telecom transceivers front-ends, a modified Volterra series approach has been adapted at system level. The approach decomposes the overall nonlinear behavior of an architecture into different contributions, each corresponding to a nonlinearity coefficient of the individual blocks that are described with a high-level model. The combination of the system's formulation into an equivalent circuit with the use of Volterra series allows to easily take into account nonlinear feedback loops and nonlinear loading effect. As illustrated with a 5 GHz WLAN receiver front-end, such analysis can be used to obtain insight in the nonlinear behavior of a system, for example to know the main contributors to the overall IP_3, or to know which block has to be made more linear to increase this IP_3. Furthermore, the knowledge of the main nonlinearities can be used to perform efficient high-level simulations by keeping the dominant nonlinear coefficients only. The use of vector calculations reduces significantly the CPU time and allows the computation of very high order responses. In this way, architectural designers quickly get an insight in the nonlinear behavior of RF front-ends.

References

[1] G. Vandersteen, P. Wambacq, Y. Rolain, P. Dobrovolny, S. Donnay, M. Engels, and I. Bolsens, "A methodology for efficient high-level dataflow simulation of mixed-signal front-ends of digital telecom transceivers," *DAC'2000*, 2000.

[2] P. Wambacq, P. Dobrovolny, S. Donnay, M. Engels, and I. Bolsens, "Compact modeling of nonlinear distortion in analog communication circuits," *DATE'2000*, pp. 350–354, 2000.

[3] S. A. Maas, *Nonlinear Microwave Circuits*. Artech House, 1988.

[4] J. Roychowdhury, "Reduced-order modeling of time-varying systems," *IEEE Trans. Circuits and Systems II*, vol. 46, no. 10, pp. 1273–1288, 1999.

[5] E. van den Eijnde, *Steady-State Analysis of Strongly Nonlinear Circuits*. VUB, 1989. PhD Thesis.

[6] E. van den Eijnde and J. Schoukens, "Steady-state analysis of a periodically excited nonlinear system," *IEEE Trans. Circuits and Systems*, vol. 37, no. 2, pp. 232–242, 1990.

[7] L. O. Chua and C.-Y. Ng, "Frequency-domain analysis of nonlinear systems: Formulation of transfer functions," *Electronic Circuits and Systems*, vol. 3, no. 6, pp. 257–269, 1979.

[8] J. J. Bussgang, L. Ehrman, and J. W. Graham, "Analysis of nonlinear systems with multiple inputs," *Proc. IEEE*, vol. 62, no. 8, pp. 1088–1119, 1974.

[9] A. A. M. Saleh, "Matrix analysis of mildly nonlinear, muliple-input, multiple-output systems with memory," *Bell Syst. Tech. Journal*, vol. 61, no. 9, pp. 2221–2243, 1982.

[10] J. W. Brewer, "Kronecker products and matrix calculus in system theory," *IEEE Trans. Circuits and Systems*, vol. CAS-25, no. 9, pp. 772–781, 1978.